(51) International Patent Classification:
G06F 11/36 (2006.01)

(21) International Application Number:
PCT/CN2005/000938

(22) International Filing Date: 28 June 2005 (28.06.2005)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant *(for all designated States except US)*: INTEL CORPORATION [US/US]; 2200 Mission College Boulevard, Santa Clara, CA 95052 (US).

(72) Inventor; and
(75) Inventor/Applicant *(for US only)*: LI, Shaofan [CN/CN]; Apt.1005, Building 39, 511 Wuzhong Road, Shanghai 200113 (CN).

(74) Agent: SHANGHAI PATENT & TRADEMARK LAW OFFICE, LLC; 435 Guiping Road, Shanghai 200233 (CN).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *with international search report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: METHOD AND SYSTEM FOR NON-INTRUSIVE CODE COVERAGE
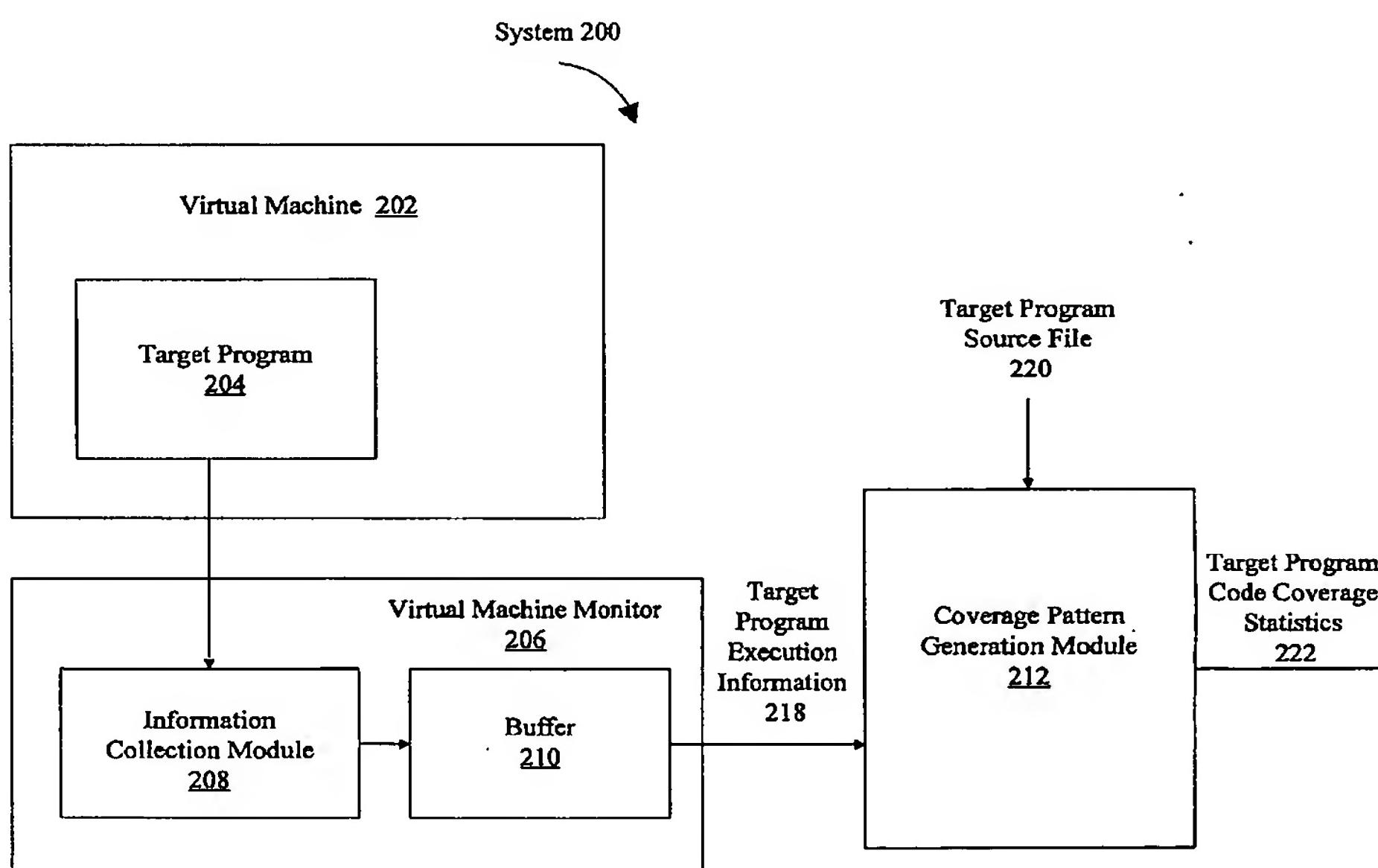
(57) Abstract: A method and system for non-intrusive code coverage is described. The method includes configuring a CPU to notify an information collection module when a branch is taken, recording a branch address when the information collection module is notified of a branch taken, and storing the recorded branch addresses to determine code coverage of a target program.

METHOD AND SYSTEM FOR NON-INTRUSIVE CODE COVERAGE

TECHNICAL FIELD

[0001]   Embodiments of the invention relate to software testing, and more specifically to non-intrusive code coverage.

BACKGROUND

[0002]   Code coverage tools are used to collect information about software testing to determine whether a software program has been tested thoroughly. Typically, code coverage tools instrument code into a target program in order to collect code coverage information. The code coverage code is either instrumented into the target program source code or into the target executable directly. This instrumentation is intrusive to the target program and increases the size of the target program.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention is illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings in which like reference numerals refer to similar elements.

FIG. 1 is a block diagram illustrating a suitable computing environment in which certain aspects of the illustrated invention may be practiced.

FIG. 2 is a block diagram illustrating one generalized embodiment of a system incorporating the invention.

FIG. 3 is a flow diagram illustrating a method according to an embodiment of the invention.

DETAILED DESCRIPTION

[0003]   Embodiments of a system and method for non-intrusive code coverage are described. In the following description, numerous specific details are set forth. However, it is understood that embodiments of the invention may be practiced without these specific details. In other instances, well-known circuits, structures and techniques have not been shown in detail in order not to obscure the understanding of this description.

[0004]   Reference throughout this specification to "one embodiment" or "an embodiment" means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment of the invention. Thus, the appearances of the phrases "in one embodiment" or "in an embodiment" in various places throughout this specification are not necessarily all referring to the same embodiment. Furthermore, the particular features, structures, or characteristics may be combined in any suitable manner in one or more embodiments.

[0005]   Fig. 1 is a block diagram illustrating a suitable computing environment in which certain aspects of the illustrated invention may be practiced. Methods of the invention may be implemented on a computer system 100 having components 102 – 112, including a central processing unit (CPU) 102, a memory 106, an Input/Output device 104, a data storage device 112, and a network interface 110, coupled to each other via a bus 108. The components perform their conventional functions known in the art and provide the means for implementing the system 100. Collectively, these components

represent a broad category of hardware systems, including but not limited to general

purpose computer systems, mobile or wireless computing systems, and specialized packet

forwarding devices. It is to be appreciated that various components of computer system

100 may be rearranged, and that certain implementations of the present invention may not

require nor include all of the above components. Furthermore, additional components

may be included in system 100, such as additional processors, storage devices, memories

(e.g. RAM, ROM, or flash memory), and network or communication interfaces.

[0006]   As will be appreciated by those skilled in the art, the content for implementing

an embodiment of the method of the invention, for example, computer program

instructions, may be provided by any machine-readable media which can store data that is

accessible by system 100, as part of or in addition to memory, including but not limited to

cartridges, magnetic cassettes, flash memory cards, digital video disks, random access

memories (RAMs), read-only memories (ROMs), and the like. In this regard, the system

100 is equipped to communicate with such machine-readable media in a manner well-

known in the art.

[0007]   It will be further appreciated by those skilled in the art that the content for

implementing an embodiment of the method of the invention may be provided to the

system 100 from any external device capable of storing the content and communicating

the content to the system 100. For example, in one embodiment of the invention, the

system 100 may be connected to a network, and the content may be stored on any device

in the network.

[0008]   Fig. 2 illustrates a system incorporating an embodiment of the invention.

System 200 includes a virtual machine (VM) 202. The virtual machine 202 may be any

platform or operating system (OS) and provides a virtualization environment to run target programs, such as 204. System 200 also includes a virtual machine monitor (VMM) 206. The virtual machine monitor 206 maintains information about the virtual machine 202 and monitors programs running on the virtual machine. The virtual machine monitor 206 includes an information collection module (ICM) 208 and a buffer 210. The information collection module 208 configures the CPU 102 to notify the information collection module 208 when a branch is taken. In one embodiment, a configuration register is set to provide a notification to the information collection module 208 when a branch is taken. In one embodiment, the CPU 102 notifies the information collection module via an interruption. When the information collection module 208 is notified by the CPU 102 about the branch, the information collection module 208 records the branch address and stores the information in the buffer 210. After runtime, the execution information 218 collected and stored in the buffer 210 may be input into a coverage pattern generation module (CPGM) 212 along with the target program source file or symbol file 220. The CPGM 212 may be any graphical user interface (GUI) that converts the target program source file 220 and the execution information 218 into code coverage statistics that may be displayed to a user.

[0009]   Fig. 3 illustrates a method according to one embodiment of the invention. At 300, a CPU is configured to notify an information collection module when a branch is taken. At 302, a branch address is recorded when the information collection module is notified of a branch taken. At 304, the recorded branch addresses are stored to determine code coverage of a target program on the CPU. In one embodiment, the recorded branch addresses are stored in a buffer. Then, after runtime, the recorded branch addresses may

be sent to a CPGM along with the target program source file to display code coverage statistics. By recording and analyzing the branches taken, the source code of the target program that was executed may be determined. This information may then aid in determining whether software testing was adequate in covering the code of the target program.

[0010]   While the invention has been described in terms of several embodiments, those of ordinary skill in the art will recognize that the invention is not limited to the embodiments described, but can be practiced with modification and alteration within the spirit and scope of the appended claims. The description is thus to be regarded as illustrative instead of limiting.

CLAIMS

What is claimed is:

1.   A method comprising:

configuring a central processing unit (CPU) to notify an information collection

module (ICM) when a branch is taken;

recording a branch address when the ICM is notified of a branch taken; and

storing the recorded branch addresses to determine code coverage of a target

program.

2.   The method of claim 1, wherein storing the recorded branch addresses

comprises storing the recorded branch addresses in a buffer.

3.   The method of claim 1, further comprising providing the recorded branch

addresses to a coverage pattern generation module (CPGM) to interpret and display code

coverage statistics.

4.   The method of claim 1, wherein the target program is to be run in a virtual

machine.

5.   The method of claim 4, wherein the information collection module is part

of a virtual machine monitor that is coupled to the virtual machine to collect code

coverage information about the target program.

6.   The method of claim 1, wherein configuring a CPU to notify an information collection module when a branch is taken comprises configuring a CPU to notify an information collection module via an interruption when a branch is taken.

7.   An apparatus comprising:

a virtual machine (VM) on which a target program is to execute; and

a virtual machine monitor (VMM) coupled to the virtual machine to gather code coverage information about the target program, the VMM including an information collection module (ICM) to configure a central processing unit (CPU) to notify the ICM when a branch is taken and to record a branch address when a branch is taken.

8.   The apparatus of claim 7, wherein the VMM further comprises a buffer to store the branch address recorded by the information collection module.

9.   The apparatus of claim 7, wherein the CPU notifies the information collection module of a branch taken via an interruption.

10.   The apparatus of claim 7, further comprising a coverage pattern generation module (CPGM) coupled to the VMM to interpret the branch addresses and a source file of the target program and to display code coverage statistics about the target program.

11.   An article of manufacture comprising:

a machine accessible medium including content that when accessed by a machine causes the machine to perform operations comprising:

configuring a central processing unit (CPU) to notify an information collection module (ICM) when a branch is taken;

recording a branch address when the information collection module is notified of a branch taken; and

storing the recorded branch addresses to determine code coverage of a target program.

12.   The article of manufacture of claim 11, wherein the machine-accessible medium further includes content that causes the machine to perform operations comprising providing the recorded branch addresses to a coverage pattern generation module (CPGM) to interpret and display code coverage statistics.

13.   The article of manufacture of claim 11, wherein storing the recorded branch addresses comprises storing the recorded branch addresses in a buffer.

14.   The article of manufacture of claim 11, wherein the machine-accessible medium further includes content that causes the machine to perform operations comprising running the target program in a virtual machine.

15.   The article of manufacture of claim 14, wherein the information collection

module is part of a virtual machine monitor that is coupled to the virtual machine to

collect code coverage information about the target program.

16.   The article of manufacture of claim 11, wherein configuring a CPU to

notify an information collection module when a branch is taken comprises configuring a

CPU to notify an information collection module via an interruption wherever a branch is

taken.

17.   A system comprising:

a processor;

a network interface coupled to the processor; and

a machine accessible medium including content that when accessed by a machine

causes the machine to perform operations including:

configuring a central processing unit (CPU) to notify an information

collection module (ICM) when a branch is taken;

recording a branch address when the information collection module is

notified of a branch taken; and

storing the recorded branch addresses to determine code coverage of a

target program.

18.   The system of claim 17, wherein the machine-accessible medium further

includes content that causes the machine to perform operations comprising providing the

recorded branch addresses to a coverage pattern generation module (CPGM) to interpret and display code coverage statistics.

19.   The system of claim 17, wherein the target program is to be run on a virtual machine.

20.   The system of claim 17, wherein the information collection module is part of a virtual machine monitor that is coupled to the virtual machine to collect code coverage information about the target program.

100

102 → CPU

104 → Memory

106 → I/O

BUS - 108

110 → Network Interfaces

112 → Data Storage

FIG. 1

System 200

Virtual Machine **202**

Target Program
204

Virtual Machine Monitor
206

Target Program Source File
220

Coverage Pattern
Generation Module
212

Target Program Code Coverage
Statistics
222

Target
Program
Execution
Information
218

Buffer
210

Information
Collection Module
208

FIG. 2

Configure a CPU to notify an
information collection module when
a branch is taken

300

Record a branch address when the
information collection module is notified
of a branch taken

302

Store the recorded branch addresses
to determine code coverage
of a target program

304
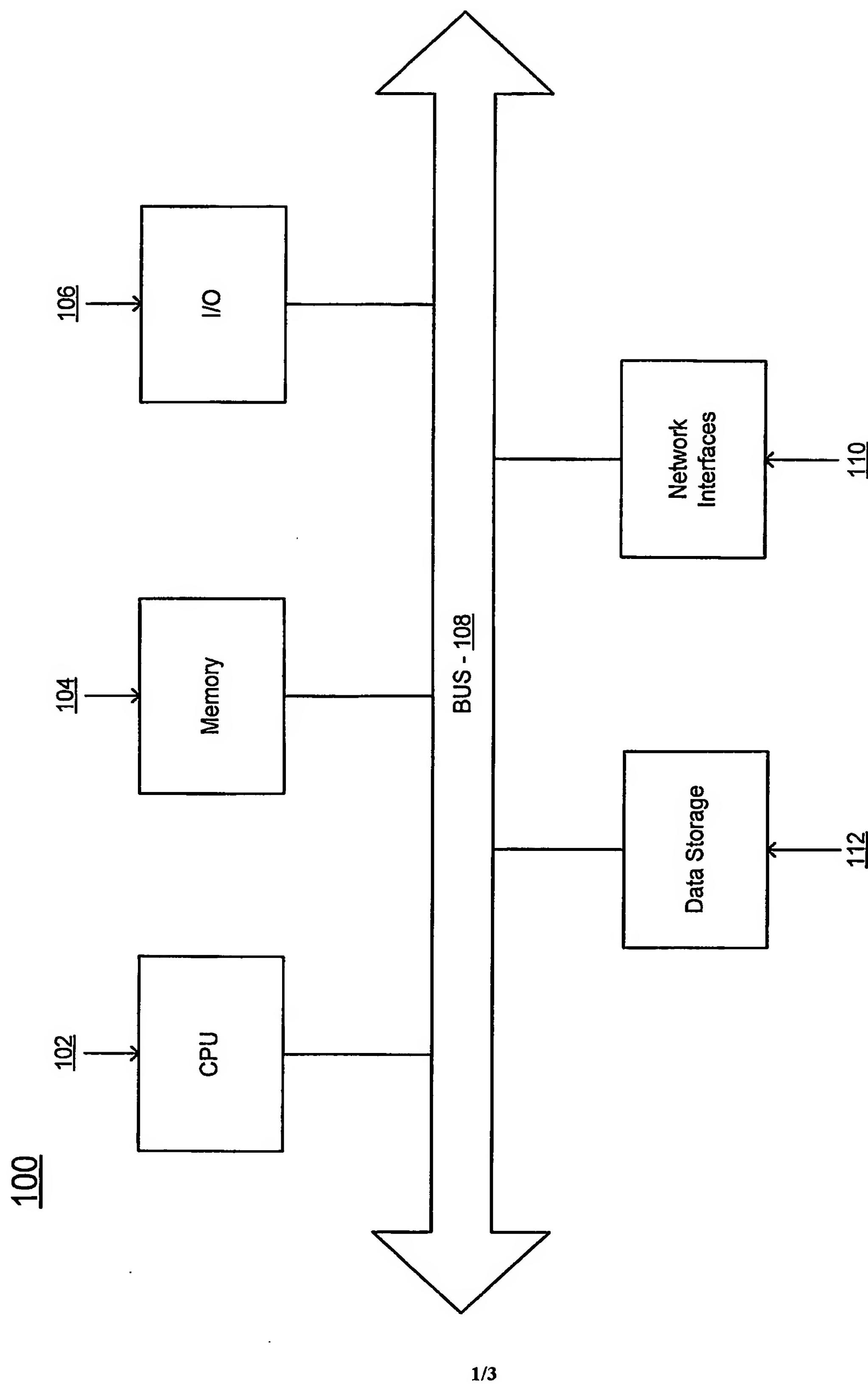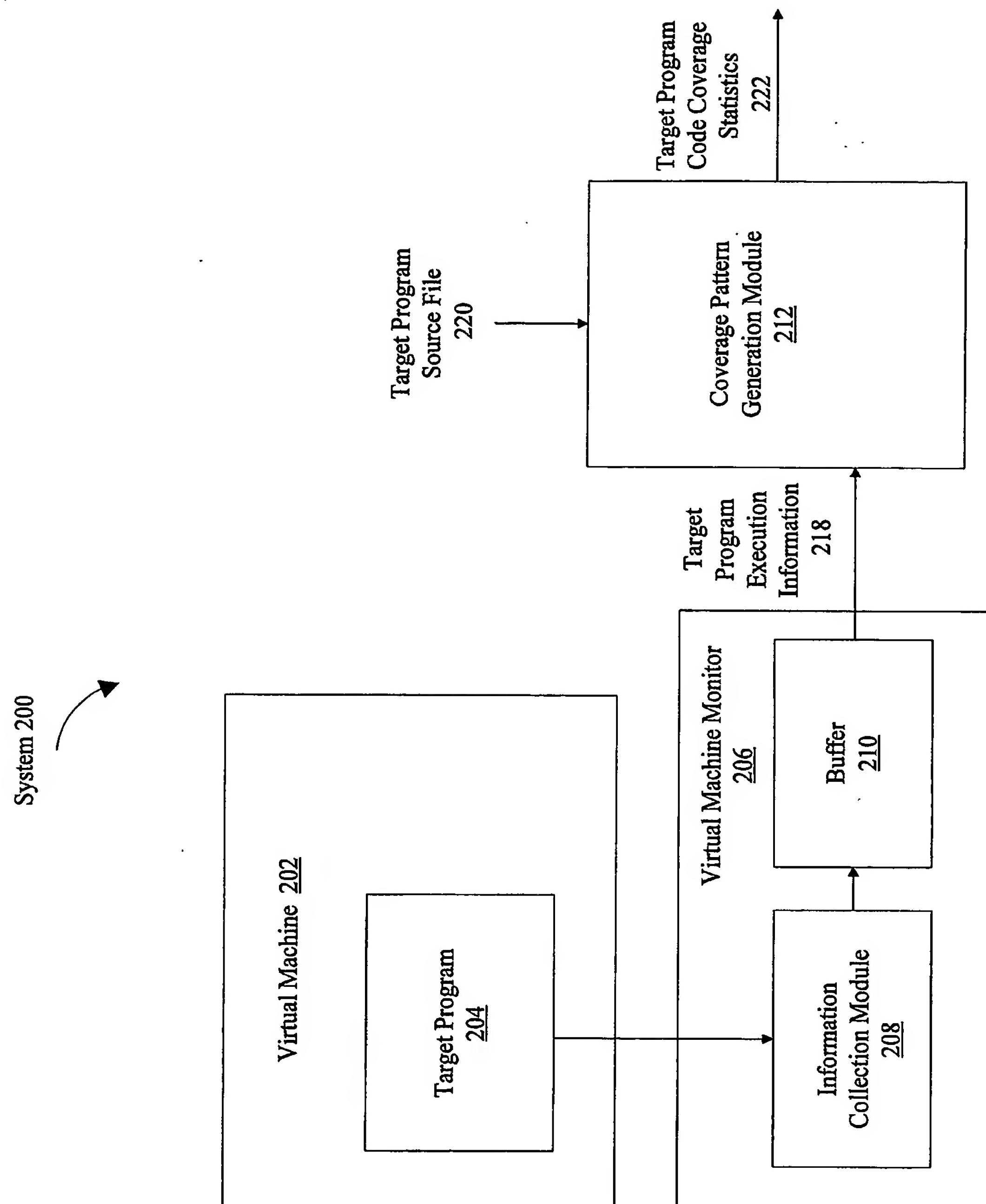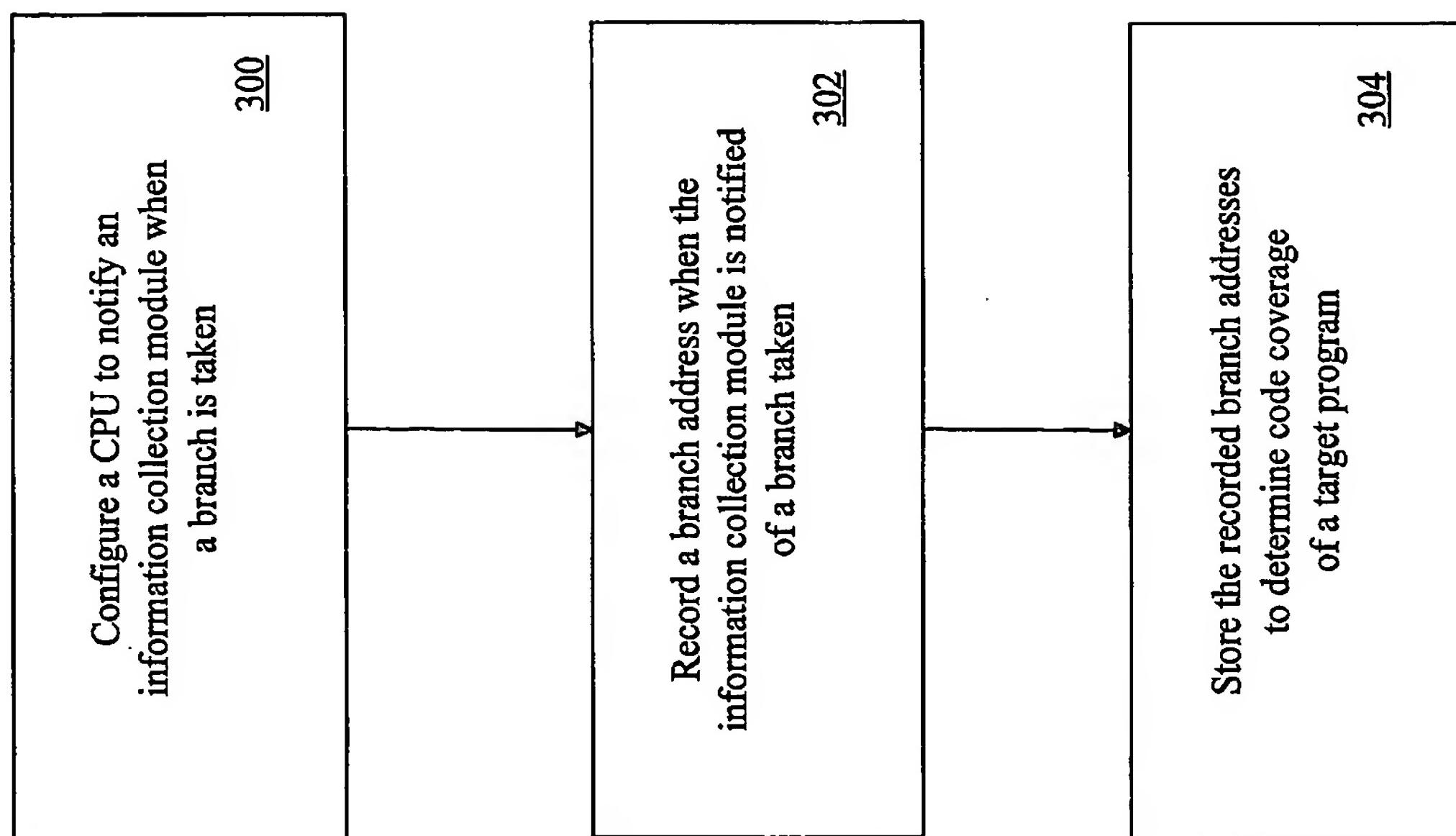
FIG. 3

INTERNATIONAL SEARCH REPORT

A. CLASSIFICATION OF SUBJECT MATTER

G06F11/36 (2006.01) i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F11/36 (2006.01) i

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CPRS, CNKI, WPI, EPODOC, PAJ

Software testing code coverage branch collecting

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	CN1630052A (NANSHANBRIDGE CO LTD) 22.Jun 2005(22.06.2005) the whole document	1-20
A	US6701519B1 (COMPUWARE CORPORATION) 02.Mar 2004(02.03.2004) the whole document	1-20
A	US6536036B1 (INTERNATIONAL BUSINESS MACHINES CORPORATION) 18.Mar 2003(18.03.2003) the whole document	1-20
A	US2003/0093716A1 (INTERNATIONAL BUSINESS MACHINES CORPORATION) 15.May 2003(15.05.2003) the whole document	1-20

☐ Further documents are listed in the continuation of Box C.   ☒ See patent family annex.

* Special categories of cited documents:	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A" document defining the general state of the art which is not considered to be of particular relevance	
"E" earlier application or patent but published on or after the international filing date	"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"L" document which may throw doubts on priority claim (S) or which is cited to establish the publication date of another citation or other special reason (as specified)	"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"O" document referring to an oral disclosure, use, exhibition or other means	
"P" document published prior to the international filing date but later than the priority date claimed	"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
01.Mar 2006(01.03.2006)	16 MAR 2006(16 · 03 · 2006)

Name and mailing address of the ISA/CN	Authorized officer
The State Intellectual Property Office, the P.R.China 6 Xitucheng Rd., Jimen Bridge, Haidian District, Beijing, China 100088 Facsimile No. 86-10-62019451	HAN Yan Telephone No. 86-10-62084929

Form PCT/ISA /210 (second sheet) (April 2005)

INTERNATIONAL SEARCH REPORT
Information on patent family members

Patent Documents referred in the Report	Publication Date	Patent Family	Publication Date
CN1630052A	22.Jun 2005(22.06.2005)	NULL	
US6701519B1	02.Mar 2004(02.03.2004)	NULL	
US6536036B1	18.Mar 2003(18.03.2003)	NULL	
US2003/0093716A1	15.May 2003(15.05.2003)	NULL	